Implementation and Algorithms for TOF, MTD and PP2PP DSM Tree – 2010 RHIC Run

Eleanor Judd

April 5, 2010

Change Log:

Date	Description
December 9, 2009	First version for 2010 AuAu run. Added 2 more TOF total multiplicity thresholds to the TF201 algorithm
December 21, 2009	Added 2 more thresholds to TF201, bringing the total to 5. Since no-one used the earlier version I just re-implemented the "a" version of this algorithm and did not change the version letter to "b".
April 5, 2010	Made the "a" version of the TF001 algorithm for 2010. Input values greater than 24 are ignored.

The only change to the algorithms in this tree for the 2010 AuAu running is to the TF201 DSM, where more TOF thresholds have been added. Everything else is as it was at the end of the 2009 run.

1. Layer 0 DSM Boards: MIX_TF001:006

The TOF layer-0 DSM boards each receive 20 5-bit multiplicity values from the TOF trays. The connections are made such that each layer-0 DSM receives TOF data from one 2-hour pie-slice of the detector. Each 5-bit number is actually a count of how many TOF trays, in a group of 24, were hit. Values greater than 24 are therefore unphysical and are ignored. The values are summed to calculate the total multiplicity.
NOTE: In order to avoid doing too many sums in parallel, this algorithm takes an extra tick of the RHIC clock, which corresponds to 4 extra ticks of the 4xRHIC clock that is used by the FPGA. This allows many sums to be performed in series, which is easier to implement.

RBT File: mix_tf001_2010_a.rbt

Users: TF001:TF006

Inputs: Ch 0:6 = TOF trays
Ch 7 = Unused

On each DSM channel:
(0:14) 3 5-bit TOF multiplicity values
(15) Unused
NOTE: Ch 6 receives just 2 input multiplicity values so it uses only bits 0:9

LUT: 1-to-1. Noisy, dead and non-instrumented channels are also zeroed out here

Registers: None

Action

1st	Latch inputs
2nd	Zero out any channel with a value greater than 24
3rd	Sum TOF channels 0:2, 3:5, 6:8, 9:11, 12:14, 15:17 and 18:19
4th	Combine these sums in pairs to make the sums of channels 0:5, 6:11 and 12:17. Delay the sum of channels 18 and 19 to the 4th clock tick
5th	Combine these sums in pairs to make the sums of channels 0:11 and 12:19
6th	Combine these two sums to make the final sums of channels 0:19
7th	Delay the final sum
8th	Latch Outputs

Output to TF101:
(0:9) TOF multiplicity
(10:15) Unused

2. Layer 1 DSM Board: MIX_TF101

The TOF layer-1 DSM board receives a 10-bit multiplicity value from each of the six TOF layer-0 DSM boards. Each input multiplicity is compared to a threshold. In parallel with this, the values are also summed to calculate the total multiplicity.
NOTE: In order to avoid doing too many sums in parallel, this algorithm takes an extra tick of the RHIC clock, which corresponds to 4 extra ticks of the 4xRHIC clock that is used by the FPGA. This allows many sums to be performed in series, which is easier to implement.

RBT File: mix_tf101_2009_a.rbt

Users: TF101

Inputs: Ch 0:5 = TF001:TF006
Ch 6:7 = Unused

On each DSM channel:
(0:9) TOF multiplicity
(10:15) Unused

LUT: 1-to-1

Registers:

R0: TOF-sector-th (10)

Action

1st	Latch inputs
2nd	Sum channels 0:1, 2:3 and 4:5 In parallel, compare each of the 6 input multiplicity values to the threshold specified in register 0.
3rd	Combine the first two sums to make the sums of channels 0:3. Delay the sum of channels 4 and 5 to the 4th clock tick. Delay the 6 threshold bits to the 8th clock tick.
4th	Combine the two remaining sums to make the final total multiplicity sum of channels 0:5.
5th	Delay the final sum to the 8th clock tick.
6/7th	No logic
8th	Latch Outputs

Output to TF201:

(0:12)	TOF total multiplicity
(13:15)	Unused
(16:21)	6 sector threshold bits
(22:31)	Unused

3. Layer 0 QT Board: MXQ_MT001

The layer 0 DSM board for the MTD has been replaced with a QT board this year. Initially this board was programmed with an algorithm that just sends good-hit bits to the DSM board. The algorithm was used by both the MTD and pp2pp QT boards. On May 13th 2009 the MTD QT board was re-programmed to use the ZDC QT algorithm, which sends good-TAC values to the DSM boards. Please see the documentation provided by Chris Perkins for a description of its algorithm.

4. Layer 1 DSM Board: MIX_MT101

The MT101 DSM board processes data from the MTD detector. Originally it also processed data from the PP2PP detector. However, during the commissioning phase it was determined that the PP2PP data was arriving too late, when compared to MTD, so on May 18th 2009 the connection to the PP2PP system was removed from this DSM board. The algorithm receives 2 “good TAC” values from the MTD QT board (MXQ_MT001). It calculates both the difference and sum of these values and then applies multiple threshold cuts to both values. The results are combined and one final bit is passed on to Layer 2.

Change Log:

Version	Date	Comment
a	03/12/2009	Original version containing both MTD and PP2PP logic.
b	05/01/2009	MTD data is actually coming in on channel 1, not channel 0, so the "Good Hit" bits were changed from input bits 0 and 4 to 16 and 20
c	05/14/2009	The MTD QT board was re-programmed to use the ZDC QT algorithm. The DSM input is now good TAC values rather than just good hit bits. The TAC difference and sum are calculated and cuts are applied. NOTE: The "c" version of this algorithm does actually contain its original PP2PP logic. However, from May 18th 2009 onward, the PP2PP system was no longer connected to this DSM so that logic was not used. The description of that logic has therefore been removed from this section of this document.

RBT File: mix_mt101_2009_c.rbt

Users: MT101

Inputs: Ch0:1 = QT Board MT001
Ch2:7 = Unused

From the MT001 QT board:
NOTE: The connections of the two cables that carry data from the MT001 QT board to the MT101 DSM are switched. As a result the 16 LSB of the QT output arrive at MT101 on channel 1 (i.e. bits 16:31) and the 16 MSB of the QT output arrive on channel 0 (i.e. bits 0:15).
bits 0:7 = 8 MSB of Good TAC value from MTD West channel (TAC-W)
bits 8:15 = Unused
bits 16:27 = Good TAC value from MTD East channel (TAC-E)
bits 28:31 = 4 LSB of Good TAC value from MTD West channel (TAC-W)

LUT: 1:1

Registers:
R0: MTD-TACdiff-Min (13)
R1: MTD-TACdiff-Max (13)
R2: MTD-TACsum-Min (13)
R3: MTD-TACsum-Max (13)

Action:
1st Latch input

2nd Calculate MTD TAC difference = 4096 + TAC-W – TAC-E
Calculate MTD TAC sum = TAC-W + TAC-E

Define: Good-TAC-E = TAC-E > 0, same for West side

3rd Compare the MTD TAC difference to its minimum and maximum values, as specified in registers 0 and 1. The logic looks for the TAC difference to be greater than the minimum and less than the maximum.
Compare the MTD TAC sum to its minimum and maximum values, as specified in registers 2 and 3. The logic looks for the TAC sum to be greater than the minimum and less than the maximum.
Combine all the MTD results to produce one final MTD bit. The TAC difference is required to be inside its window, but the TAC sum must be outside its window:

MTD = Good-TAC-E and Good-TAC-W and
(reg0 < TAC-difference < reg1) and
not (reg2 < TAC-sum < reg3)

4th Latch output

Output to TF201:
(0) MTD
(1:15) Unused

Scalers:
Unused

5. Layer 0 QT Board: MXQ_PP001

The layer 0 DSM board for the PP2PP detector has been replaced with a QT board this year. Please see the documentation provided by Chris Perkins for a description of its algorithm. Initially the output of this QT board was connected to the input of the MT101 DSM. However, during commissioning of the PP2PP detector it was determined that the data arrived too late, when compared to MTD. So, on May 18th 2009 the QT board output cables were moved from MT101 to TF201. It was subsequently determined that the timing of this connection was good (i.e. the PP2PP data arrived at the same time as the TOF and MTD data) so the cables were left in place and the TF201 DSM algorithm was modified accordingly.

6. Layer 2 TOF DSM Board: L1-TF201

All the information from the TOF, MTD and PP2PP detectors is brought into the TOF layer 2 DSM. The single MTD bit and the TOF sector threshold bits are simply passed through to the last DSM or new TCU. The TOF multiplicity is compared to three thresholds. The algorithm also receives 16 “good hit” bits from the PP2PP QT board (MXQ_PP001). There is one bit from each of 16 PMTS. The bits are combined in pairs to make a bit for each Roman Pot, and then the RP bits are combined to make the components of elastic and inelastic triggers. The PP2PP bits are combined to make the elastic and inelastic trigger bits.

Change Log:

Version	Date	Comment
2009_a	03/12/2009	Original version containing MTD and half the PP2PP logic but no TOF logic.
2009_b	05/28/2009	Added in the TOF logic. The first half of the PP2PP logic, which used to be in the MT101 DSM, was added into this algorithm.
2010_a	12/09/2009	Added in two more TOF total multiplicity thresholds
2010_a	12/21/2009	Added in two more TOF total multiplicity thresholds. No-one used the 12/09/2009 version, so I just modified it instead dof making a "2010_b" version.

RBT File: l1_tf201_2010_a.rbt

Users: TF201

Inputs: Ch 0 = MT101
Ch 1 = Unused
Ch 2:3 = TF101
Ch 4 = PP001 (QT Board)
Ch 5:7 = Unused

From MTD Layer 1 DSM - MT101
(0) MTD
(1:15) Unused

From TOF Layer 1 DSM - TF101
(0:12) TOF total multiplicity
(13:15) Unused
(16:21) TOF sector threshold bits
(22:31) Unused

From the PP001 QT board: "Good Hit" bits from 16 PMTS
bit 0:3 = RPEVU1, RPEVU2, RPEVD1, RPEVD2
bit 4:7 = RPWVU1, RPWVU2, RPWVD1, RPWVD2
bit 8:11 = RPEHO1, RPEHO2, RPEHI1, RPEHI2
bit 12:15 = RPWHO1, RPWHO2, RPWHI1, RPWHI2

LUT: 1-to-1

Registers:
R0: TOF-Mult-th0 (13)
R1: TOF-Mult-th1 (13)
R2: TOF-Mult-th2 (13)
R3: TOF-Mult-th3 (13)
R4: TOF-Mult-th4 (13)

Action

1st Latch inputs

2nd Delay the MTD bit and the TOF sector threshold bits to the 4th step.
Compare the TOF total multiplicity to the thresholds specified in registers 0 to 4.
Combine (OR) the 16 PP2PP good hit bits to make the 8 Roman Pot (RP) bits, i.e.:

EVU = RPEVU1 or RPEVU2
EVD = RPEVD1 or RPEVD2
WVU = RPWVU1 or RPWVU2
WVD = RPWVD1 or RPWVD2
EHO = RPEHO1 or RPEHO2
EHI = RPEHI1 or RPEHI2
WHO = RPWHO1 or RPWHO2
WHI = RPWHI1 or RPWHI2

Then combine the Roman Pot bits to make the elastic and inelastic trigger components, i.e.:

EA = WVU and EVD
EB = WVD and EVU
EC = WHO and EHI
ED = WHI and EHO
EOR = EVU or EVD or EHO or EHI
WOR = WVU or WVD or WHO or WHI
EVF = EVU and EVD
EHF = EHI and EHO
WVF = WVU and WVD
WHF = WHI and WHO

3rd Delay the TOF multiplicity bits and the 10 PP2PP trigger component bits to the 4th step.
In parallel, combine those PP2PP component bits to make the raw trigger conditions and their vetoes, i.e.:

ET_raw = EA or EB or EC or ED
ET_veto = WVF or WHF or EVF or EHF
ITE_raw = EOR
ITE_veto = EVF or EHF
ITW_raw = WOR
ITW_veto = WVF or WHF

Finally, combine each raw trigger bit with its veto to make the PP2PP trigger bits, i.e.:

ET = ET_raw and not ET_veto
ITE = ITE_raw and not ITE_veto
ITW = ITW_raw and not ITW_veto

4th Latch Outputs

Output to TCU:

Bit	Name	Description
Bit 0	MTD	MTD trigger
Bit 1	ET	PP2PP elastic trigger
Bit 2	ITE	PP2PP East inelastic trigger
Bit 3	ITW	PP2PP West inelastic trigger
Bit 4	TOFmult0	TOF total multiplicity > th0
Bit 5	TOFmult1	TOF total multiplicity > th1
Bit 6	TOFmult2	TOF total multiplicity > th2
Bit 7	TOFmult3	TOF total multiplicity > th3
Bit 8	TOFmult4	TOF total multiplicity > th4
Bit 9	TOFsector0	TOF sector 0 multiplicity > th
Bit 10	TOFsector1	TOF sector 1 multiplicity > th
Bit 11	TOFsector2	TOF sector 2 multiplicity > th
Bit 12	TOFsector3	TOF sector 3 multiplicity > th
Bit 13	TOFsector4	TOF sector 4 multiplicity > th
Bit 14	TOFsector5	TOF sector 5 multiplicity > th
Bit 15	N/A	Unused

Output to Scalers:

Bit	Description
Bit 0	EA
Bit 1	EB
Bit 2	EC
Bit 3	ED
Bit 4	EOR
Bit 5	WOR
Bit 6	EVF
Bit 7	EHF
Bit 8	WVF
Bit 9	WHF
Bit 10	MTD
Bit 11	TOF total multiplicity > th0
Bit 12	TOF total multiplicity > th1
Bit 13	TOF total multiplicity > th2
Bit 14	TOF total multiplicity > th3
Bit 15	TOF total multiplicity > th4